Alternatives for Classifier System Credit Assignment

Gunar E. Liepins
Michael R. Hilliard
Oak Ridge National Laboratory
PO Box 2008, MS 207, Bldg. 4500N
Oak Ridge, Tennessee 37831-6207

Mark Palmer
Gita Rangarajan
Energy Environment and Resources Center
10521 Research Drive, Suite 100
Knoxville, Tennessee 37932

Abstract

Classifier systems are production rule systems that automatically generate populations of rules cooperating to accomplish desired tasks. The genetic algorithm is the systems' discovery mechanism, and its effectiveness is dependent in part on the accurate estimation of the relative merit of each of the rules (classifiers) in the current population. Merit is estimated conventionally by use of the bucket brigade for credit assignment. This paper addresses the adequacy of the bucket brigade and provides a preliminary exploration of two variants in conjunction with enumerated rules and with discovery. In limited experiments, a variant that combines the bucket brigade, "classifier chunking," and "backwards averaging" has yielded improved performance on simple maze problems. Tentative similarities between this hybrid and Sutton's Adaptive Heuristic Critic (AHC) are suggested.

Area B: Fundamental Problems, Methods, Approaches Subarea B2: Knowledge Acquisition, Learning, Analogy

1. Introduction

Credit assignment is the problem of determining how to reinforce individual rules in a multistep chain when the external reward is given only at the chain's conclusion. Some of the earliest work in credit assignment was Samuel's celebrated checker-playing program (1959 and 1967) which used a heuristic version of temporal difference (TD) methods (Sutton, 1988). These methods are similar in philosophy to the Adaptive Heuristic Critic (AHC) developed by Sutton (1984) and Barto, Sutton and Anderson (1983), the bucket brigade developed by Holland (1985 and 1986), and the learning systems studied by Witten (1977), Booker (1982), and Hampson (1983). Each of these provides a mechanism whereby adjustments to rule strength are made in an incremental fashion, in contrast to supervised learning with various backwards averaging schemes (Grefenstette, 1988; Widrow and Stearns, 1985; Holland and Reitman, 1978; and Rumelhart, Hinton and Williams, 1986).

This paper studies credit assignment in an environment with minimal prior knowledge. Initial investigation of three credit assignment methods has been undertaken in terms of their speed and accuracy in learning abstract state-space maze problems. The three methods studied are the bucket brigade, backwards averaging with classifier chunking, and a combination of the methods. The correspondence to TD and AHC methods is suggested.

1.1 Classifier Systems

Classifier systems to automatically discover rules to perform desired tasks were developed by Holland and Reitman (1978) and later refined by Holland (1986). In contrast to traditional expert systems where rules are handcrafted by knowledge engineers, classifier systems use the genetic algorithm as a discovery operator to generate rules. Each classifier is an "if-then" rule, with a condition part and an action part. A message list is used to store the current environmental state and any internal messages. Associated to each classifier is a numerical value called its strength. Holland and Reitman (1978) adjusted classifier strength through backwards averaging and other central methods. The current credit assignment standard is the bucket brigade (Holland, 1985).

Should the conditional part of a classifier match a message(s) on the message list, the classifier pays a portion of its strength (its bid) for the privilege of acting, whereupon the consequent action (which may be either an explicit action or the posting of an internal message) is taken. In the simplest classifier system implementation, the bid payment made by the acting classifier(s) is paid in equal proportions to classifiers acting in the previous cycle. Should competing classifiers specify incompatible actions, conflict resolution is based on the magnitude of the effective bids (determined as the sum of the bid plus a random variable chosen from a distribution determined by a noise schedule). If the action results in an evaluation state, an appropriate reward or punishment is assigned to the acting classifiers of the current cycle. The "bid-payment" cycle is repeated for a predetermined number of cycles, whereupon the individual classifiers contribute to a gene pool in direct proportion to their strength and the genetic recombination operators are invoked.

Since Holland and Reitman's work, several variants of classifier systems have been successfully demonstrated including solution of a difficult Boolean discovery problem (Wilson, 1987a), discovery of an optimal pumping schedule and automatic leak detection for gas pipelines (Goldberg, 1983), and discovery of probabilistic scheduling rules for job shop scheduling problems (Hilliard et. al., 1988). For additional descriptions of classifier system applications, the reader is referred to Davis, 1987; and Goldberg, 1989.

1.2 Problem Selection

The test bed of problems for this study were one, two, and three dimensional mazes with 64 possible states and with a specified start and goal. The chosen representation encoded the states as lattice points in Euclidean space. Allowable actions at any state were to make a single step move parallel to the coordinate axes (subject to the constraint that the move did not go outside the lattice). Upon attainment of the goal, the system was given a reward R and was reset to the start.

Certain of the problems incorporated "barriers" at selected non-goal states. In one formulation, barrier states were all equally resistant to entry; in another, the barriers had relative "holes," states less resistant to entry. Barriers were not explicitly represented; instead, classifiers responsible for system entry into barrier states had their strength immediately decremented. This challenged the system to learn to associate certain states with punishment either through internally developed messages, mapping expected rewards onto states, or simply through the credit assignment mechanism. The purpose of the barriers was to provide an experimental test to determine whether classifier systems could learn to pass through undesirable states (local rninirnums) to attain a global goal state (global maximums), and whether the systems could learn to distinguish between undesirable states of different intensities.

2. Credit Assignment

Although even genetic algorithm and stimulus-response classifier system performance is affected by the reward structure (Wilson, 1987a), reward and credit allocation issues become most apparent when rewards are delayed. Such a delay in reward causes difficulties for the genetic algorithm (discovery) stage of the classifier system, since its success depends on accurate estimates of the relative merit of the classifiers. Proper assessment of merit requires that the system frequently attain evaluation states and that the rewards and punishments be properly distributed to earlier stage setting classifiers. This must be accomplished rapidly without improperly restricting system exploration.

2.1 Earlier Studies

Credit assignment for "stage setting" classifiers has already been investigated by Wilson (1987b), Grefenstette (1988), and Riolo (1987,88). Wilson and Riolo's studies have been limited, effectively, to non-competitive chains of classifiers (Riolo had one pair of competing classifiers) and did not study credit assignment in conjunction with genetic discovery. Results suggest that the number of steps necessary to reinforce a classifier in a chain (i.e., the number of cycles until classifier-strength had attained a fixed proportion of steady state strength) is a linear function of the number of steps that the classifier is removed from the reward state.

Sometimes, even a linear function of the number of steps removed from the goal may be too slow for useful classifier chain development—Riolo's results suggested that some 2100 cycles were required for attainment of 90% of steady state for a classifier 12 steps removed from the goal. This potential difficulty has been recognized by Holland (1985) who conjectured that "bridging classifiers" would accelerate allocation of credit. Riolo (1987,88) has undertaken experiments that substantiate the potential usefulness of "bridging classifiers" once they have been discovered or manually injected. Unfortunately, the authors know of no case where "bridging classifiers" have been discovered by the system (our experiments have discovered rule sets in which certain classifiers act at more than one point in a rule chain and speed the distribution of credit, but they are not of the form Riolo suggests and their development will depend on the chosen representation). To address the apparent failure of the bridging classifiers to spontaneously develop, Wilson (1987b) has formulated the concept of hierarchical credit allocation, but that concept remains untested.

In contrast to Wilson and Riolo, Grefenstette (1988) was interested in predictive accuracy of learning and not learning speed. He compared RUDI, a variant of LS-1 (Smith, 1980), against the classifier system with the bucket brigade. He also provided experimental evidence that the bucket brigade leads to learning the next classifier's strength, whereas PSP (the profit sharing plan — backwards averaging with no attenuation) learns the expected system reward along the current subchain.

This study differs from previous work insofar as it addresses speed of learning of optimal solutions (shortest paths start to goal; and in the case of barrier with hole, passing through the hole), both with fully enumerated conflicting rules and with discovery.

2.2 Desiderata and Techniques

Three heuristics motivate the credit assignment methods studied: 1. Reward the rules that have acted to form the chain from the start to the goal. 2. Draw the reward back from the goal to the start. 3. Break cycles.

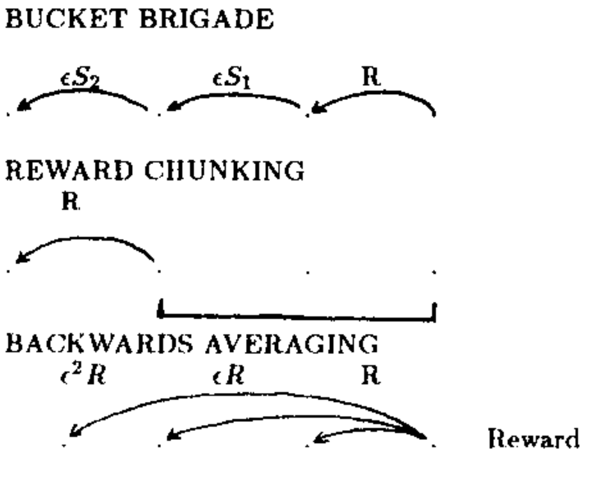
Classifier reward is accomplished with three different mechanisms in this study: First, the conventional bucket brigade. Second, a geometrically attenuated backwards averaging of the reward with classifier chunking. Third, a combination of bucket brigade, backwards averaging, and classifier chunking. Backwards averaging is implemented through the maintenance of a list of the classifiers that have fired since the last initialization of the task to the start. Once the reward is attained each classifier in the chain is rewarded by the factor λ^k, where the term A,0 < A < 1, is the attenuation factor and k represents the number of steps between the last firing of the classifier and the attainment of the reward.

Classifier chunking passes the full reward (without attenuation) to all classifiers that have achieved a given percentage of steady state and link to the goal through (minimum subpath) steady state classifiers. Any additional attenuation is begun from this point. Cycle breaking (preventing "infinite loops" which do not produce useful actions) is accomplished by the assessment of an action tax. (This action tax has one additional benefit; it encourages the development of short chains.) These concepts are illustrated in Figure 1, where

$S_i(t)$ = strength of classifier i at updating t

ϵ = bid constant

R = reward



$$S_i(t+1) = S_i(t) + \epsilon[S_{i-1}(t) - S_i(t)], i \neq 0$$
$$S_o(t+1) = S_o(t) + \epsilon[R/\epsilon - S_o(t)]$$

$$R_i(t) = \begin{cases} R & \text{If classifier i results in a ``chunked''} \\ & \text{reward state, otherwise credit cal-} \\ & \text{culated as if reward were at the earliest} \\ & \text{classifier in the ``chunked'' subchain} \end{cases}$$

Figure 1. Three Credit Assignment Mechanisms

3. Experiments

The environments chosen for the experiments reported here are 8 x 8 mazes. The mazes were used to verify the (sub) linearity of the bucket brigade credit allocation scheme as reported previously by Wilson (1987b) and Riolo (1987) and to compare the three credit allocation schemes — unmodified bucket brigade, backwards averaging with classifier chunking, and combination with classifier chunking.

3.1 Implementation

For every cycle of each maze, the classifier system is provided with the current state coordinates. The system matches these coordinates against the current population of rules and, through a stochastic conflict resolution scheme based on rule strengths and noise schedule, determines the next action. The action is taken, and the system is moved to the new state. Since the state space is small (64 cells) in each of the mazes, all possible specific rules can be enumerated. If general rules (rules matching multiple states) are considered, then the size of the rule set becomes large under full enumeration and discovery becomes an important mechanism. In the discovery mode, the initial populations of classifiers were randomly chosen, and the genetic algorithm modified these populations as the search progressed. If at some cycle, the current state matched the conditional part of no classifier in the population, the "cover detector" mechanism (Robertson and Riolo, 1988) was invoked. The cover detector generates a classifier that matches the current state by copying the state into the classifier's conditional side, randomly flipping some of the bits to "don't cares" and randomly choosing the consequent action.

3.2 Parameter Settings

Each system was rewarded 500 units when it reached the goal, and was penalized -100 units if it suggested an action which would take it out of the maze (i.e. it bumped into the wall—a system that bumps into a wall does not change its environmental state, it is simply penalized). A penalty of -100 was associated to each cell of the barrier. The bucket brigade used a bid constant of 0.1, that is, each rule that acts pays one tenth of its strength to the previously acting rule. An action tax of .01 was also assessed to discourage closed cycles and encourage shorter chains of rules. In contrast to the local payments of the bucket brigade, backwards averaging only occurs when the goal was reached. (The attenuation factor was 0.5. Moreover, each rule is rewarded only once, according to its most recent firing in the chain—the recency heuristic, but not the frequency heuristic.) Other parameter settings were tried; these seemed to provide the best compromise between speed and finding an optimal solution.

Classifier chunking "chunked" the reward back through the maze as classifiers achieved 70% of steady state strength and linked to (minimal path) steady state subchains to the goal. Finally, the combination method used both the bucket brigade and backwards averaging.

4. Results

For the experiments with the bucket brigade and enumerated rules, steady state strengths were analytically calculated. In all other cases, a rule was assumed to be at steady state when the net gain or loss in strength throughout the traversal of the path was less than 5% of the current strength. In all cases, credit allocation was observed to be faster than a linear function of the distance to the goal as previously noted by Wilson(1987b) and Riolo (1987) in simpler experiments. (Although it might seem that according to the empirical definition of steady state a rule could achieve steady state at one cycle and lose it on another, that was never observed to happen.) It was also observed that 70% of steady state was a conservative measure of chain development; repeatable (sub) chains were observed to develop far earlier.

The remaining results are highly tentative. Full experimental designs weren't run; parameter settings weren't systematically studied, nor were all combinations of methods investigated. A more systematic study will be undertaken.

4.1 Matrix Maze Problems-Enumerated Rules

Backwards averaging alone or in combination with the bucket brigade provides much faster feedback to the system than the bucket brigade alone. The barrier invariably slowed the learning for all the credit allocation mechanisms studied. (See Figure 2.) A potential drawback to more rapid credit assignment is the loss of exploration. With backwards averaging or the combination method, the system sometimes settled on a chain of rules longer than the minimal 14 steps. Also, in some limited experiments with the barrier with hole, the bucket brigade found the hole while the backwards averaging and combination runs tended to cross at higher penalties.

4.2 Matrix Maze Problems-Discovery

In comparison to the system with enumerated (specific) rules, convergence was much more rapid for the system with discovery of general rules. Presumably, the speed up is attributable to generalization and not to discovery, although follow-up experiments to confirm this hypothesis have not been carried out. The ability to use a rule multiple times within a chain both shortens the effective length of the chain and provides earlier feedback to stage setting rules near the start. The bucket brigade can pass strength from a successor of a rule's instance late in the chain to the predecessor of the rule's instance early in the chain; therefore, the convergence curves are no longer monotonic. The pure backwards averaging method was not successful in finding minimal paths in the discovery environment, so that method is omitted from the results. Moreover, none of the methods was highly successful with the barrier with hole problem.

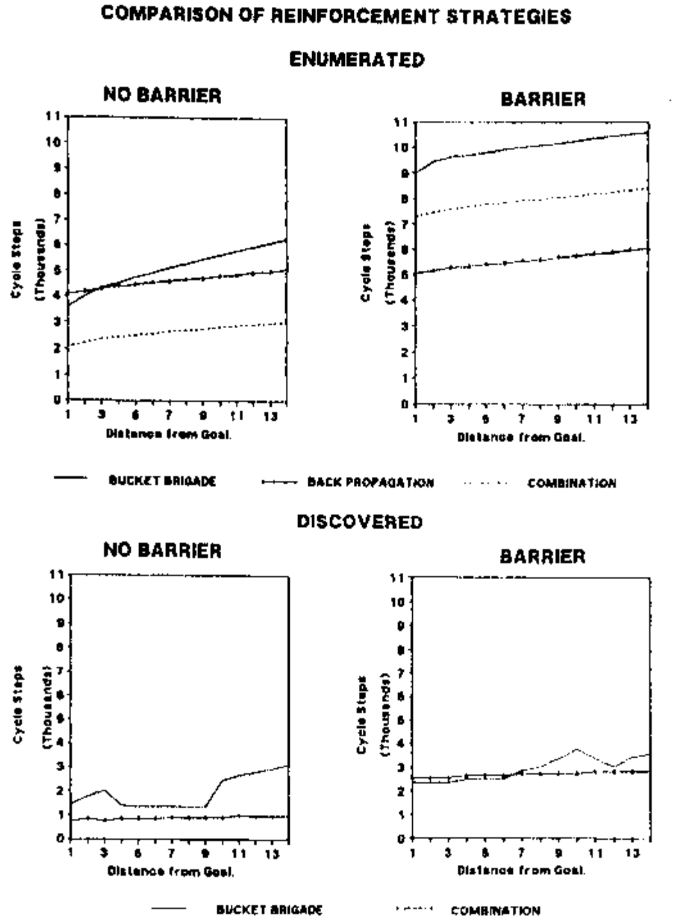


Figure 2. Steady state results for enumerated and discovered rules in a two-dimensional environment.

The bucket brigade credit assignment method encouraged sets of rules which navigated the maze efficiently and avoided bumping into the walls, a serious problem for the combination method. Using the bucket brigade, a hierarchy developed in which rules specific to rows and columns adjacent to the walls dominated more general rules specifying movement down or to the right.

5. Relationship to Other Problems and Approaches

The credit assignment problem is of wide interest in all machine learning research that deals with delayed rewards, as for example, game playing. The approach developed here is closely related to Sutton's work leading to the Adaptive Heuristic Critic and the Method of Temporal Differences. In fact, the AHC approach currently is being incorporated into the classifier system framework as an extension of this work (Brumgard, 1988).

5.1 Adaptive Heuristic Critic (AHC)

The mechanism of the adaptive heuristic critic (AHC) can be directly incorporated into the classifier system and generates counterpart behavior to the heuristic credit allocation mechanisms explored in sections 3 and 4 of this paper. Let *x[t]* be a (dimension 64) vector representation of the system state at time t; *Xi(t)* = 1 if the system is in state t at time *t*, *Xj(t)* = 0 otherwise. Let *c[t]* be the indicator vector for classifier firing; c,(2) = 1 if classifier *i* fires at time *t*, and *Cj(t)* = 0 otherwise. Let $S_c[t]$ be the classifier strength vector at time t, and *T* a tax. R[t] represents the environmental reward (positive, negative, or zero) received at state t. The predicted reward $p^s[t]$ represents the prediction (made while in state *s)* of the reward at state *t*, the heuristic reward *r[t* + 1] estimates the net gain (after the tax *T)* of moving from state *t* to state *t* + 1. The vector *v[t]* is the reward association vector at the time *t* and provides the current estimate of the expected reward at each of the states. Equation (5) defines the generic trace operator; the discounted average of past state visitations or classifier firings. The appropriate equations for the AHC formulation of the maze problems studied earlier in this paper are those for tirne-until-success tasks.

$$p^s[t] = \sum_{i=1}^{64} v_i\ [s]\ x_i\ [t] \tag{1}$$

$$\hat{r}[t+1] = -T + p^t\ [t+1]\ - p^t[t] + R[t] \tag{2}$$

$$v\ [t+1] = v\ [t] + \beta\ \hat{r}\ [t+1]\ \bar{x}\ [t] \tag{3}$$

$$S_c\ [t+1] = S_c\ [t] + \alpha\ \hat{r}\ [t+1]\ \bar{c}\ [t] \tag{4}$$

$$\text{where } \bar{z}\ [t] = (1-\lambda) \sum_{k=0}^{t-1} \lambda^k\ z\ [t-k], \quad \text{for } z = x \text{ or } z = c. \tag{5}$$

Commonalities with the previously described approach include the following: 1. The term "- T" in the heuristic reward plays the counterpart role to a tax in the classifier system. 2. The heuristic reward *r[t* + 1] estimates the net gain (after tax T) of moving from state t to t + 1. (The bucket brigade is the classifier counterpart to this, but for rules rather than states). 3. The strength update equation (4) affects a geometrically attenuated backwards averaging of reward, both from the goal state, and from the heuristic rewards. (The classifier formulation in sections 3 and 4 did not use a full trace and did not incorporate an heuristic reward into strength updates). 4. The reward association vector *v[t]* in effect "draws the reward back," somewhat analagously to classifier chunking.

There are important differences between the combination method and the AHC. The AHC estimates expected reward at a state independent of classifier strengths, allowing the tradeoff between exploration and exploitation to be better controlled. Small values for the learning parameter encourage exploration; large values encourage exploitation (Sutton, 1989).

Another difference is that classifiers are loosely linked by equation (4) of the AHC formulation and strongly linked by the bucket brigade. Preliminary considerations suggest that an AHC - bucket brigade combination could provide a "second differences" effect that could help traverse local minima.

5.2 AHC Success

Brumgard (1988) reports success with limited experiments using AHC with the square maze. He reports performance comparable to the combination method in all fully enumerated cases and the discovery barrier/no barrier cases. For the discovery barrier with hole case his AHC implementation was unable to find the hole. Sutton (1989) used AHC to solve a different (but similar) planar maze problem with barrier and hole (with enumerated rules) wherein any suggested move into the barrier was reset to the previous state and the only effect on the AHC system (l)-(5) was the incurrence of the single-step tax "-T" and increment in time.

6. Conclusions

The incorporation of backwards averaging and classifier chunking into the classifier system seems to speed convergence; however, in the case of general rules (with discovery), this speed up may sacrifice exploration and generate less than optimal rule sets. Whether this is a property of general rule sets alone, or if discovery somehow plays a role, is not known. Nor is it known if the various parameters could be tuned to overcome these difficulties. In contrast, the bucket brigade does demonstrate the ability to assign credit in (sub) linear time, and seems to be especially competitive in the discovery setting with generalized rules and a barrier. Theoretical considerations and limited experiments suggest that adaptive heuristic critic holds promise for hybrid classifier systems. Finally, the acknowledgement is made that these represent a highly preliminary set of experiments and much remains to be learned about credit assignment.

ACKNOWLEDGMENTS

The authors wish to thank Richard Sutton for many interesting and fruitful discussions.

REFERENCES

[Barto et al., 1983] A. G. Barto, R. S. Sutton and C.W. Anderson. Neuronlike Elements That Can Solve Difficult Learning Control Problems, IEEE Trans, on Systems, Man, and Cybernetics, SMC-13, 834-846.

[Booker, 1982] L. B. Booker. Intelligent Behavior as an Adaptation to the Task Environment, Ph.D. Thesis, University of Michigan, Department of Computer and Communications Sciences, Ann Arbor, MI.

[Brumgard, 1988] D. E. Brumgard. Temporal Difference Methods for Credit Allocation in Classifier Systerns, University of Tennessee, Computer Science Department Class Project, Knoxville, TN.

[Davis, 1987] L. Davis, (ed) Genetic Algorithms and Simulated Annealing, Pittman Press, London.

[Goldberg, 1983] D. E. Goldberg. Computer-aided Gas Pipeline Operation Using Genetic Algorithms and Machine Learning, Ph.D. Thesis, University of Michigan, Department of Civil Engineering, Ann Arbor.

[Goldberg, 1989] D. E. Goldberg. Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley.

[Grefenstette, 1988] J. J. Grefenstette. Credit Assignment in Rule Discovery systems Based on Genetic Algorithms, Machine Learning (8)213, 225 246.

[Hampson, 1983] S. E. Hampson. A Neural Model of Adaptive Behavior, Ph.D. Dissertation, Department of Information and Computer Sciences, University of California, Irvine.

[Milliard et al., 1988] M. R. Hilliard, G. E. Liepins, M. Palmer and D. J. Kejitan. Machine Learning Applications to Job Shop Scheduling, Proceedings of the First International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, ACM Press.

[Holland, 1985] J. II. Holland. Properties of the Bucket Brigade, Proceedings of the First International Conference on Genetic Algorithms and their Applications, Lawrence Erlbaum, Hillsdale, New Jersey, 1-7.

[Holland, 1986] J. H. Holland. Escaping Brittleness: The Possibilities of General-Purpose Learning Algorithms Applied to Parallel Rule-Based Systems, in R. S. Michalski, J. G. Carbonell, and T. M. Mitchell (eds.), Machine Learning: An Artificial Intelligence Approach, vol.2, Morgan Kaufman, Los Altos, CA.

[Holland and Reitman, 1978] J. H. Holland and J. S. Reitman. Cognitive Systems Based on Adaptive Algorithms, in D.A. Waterman and F. Hayes-Roth (eds), Pattern-Directed Inference Systems, Academic Press, New York, NY.

[Riolo, 1987a] R. L. Riolo. Bucket Brigade Performance: I. Long Sequences of Classifiers, in Genetic Algorithms and Their Applications, Proceedings of the Second International Conference on Genetic Algorithms, J.J. Grefenstette, ed., Lawrence Erlbaum Associates, Hillsdale, New Jersey, 184-195.

[Riolo, 1988] R. L. Riolo. Empirical Studies of Default Hierarchies and Sequences of Rules in Learning Classifier Systems, Ph.D Thesis, University of Michigan, Computer Science and Engineering Department, Ann Arbor, ML

[Robertson and Riolo, 1988] G. G. Robertson and R. L. Riolo. A Tale of Two Classifier Systems, Machine Learning (3) 213, 139-160.

[Rumelhart et al, 1986] D. E. Rumelhart, G. E. Hinton, and R. J. Williams). Learning Internal Representations by Error Population in Rumelhart, D. E., J. L. McClelland and the PDF Research Groups, Parallel Distributed Processing, MIT Press, 318-364.

[Samuel, 1959] A. L. Samuel. Some Studies in Machine Learning Using the Game of Checkers, IBM Journal on Research and Development, 3, 211-229.

[Samuel, 1967] A. L. Samuel. Some Studies in Machine Learning Using the Game of Checkers II - Recent Progress, IBM Journal of Research and Development. 11, 601-617.

[Smith, 1980] S. F. Smith. A Learning system Based on Genetic Adaptive Algorithms, Ph.D. Thesis, University of Pittsburgh, Computer Science Department, Pittsburgh, PA.

[Sutton, 1984] R. S. Sutton. Temporal Credit Assignment in Reinforcement Learning, Ph.D. dissertation, University of Massachusetts, Amherst, MA.

[Sutton, 1988] R. S. Sutton. Learning to Predict by the Methods of Temporal Differences, Machine Learning (2) 1, 9-44.

[Sutton, 1989] R. S. Sutton. Personal Communication.

[Widrow and Stearns, 1985] B. Widrow and S. D. Stearns. Adaptive Signal Processing. Prentice-Hall, Englewood Cliffs, New Jersey.

[Wilson, 1987a] S. W. Wilson. Classifier Systems and the Anirnat Problem, Machine Learning, 2, 199-228.

[Wilson, 1987b] S. W.Wilson. Hierarchical Credit Allocation in a Classifier System in Davis, (ed), Genetic Algorithms and Simulated Annealing Pittman Publishing, pp. 14-115.

[Witten, 1977] I. H. Witten. An Adaptive Optimal Controller for Discrete-Time Markov Environments, Information and Control, 34, 286-295.